

PCTWORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

| | | |
|--|-----------|--|
| (51) International Patent Classification ⁷ : G06F 17/30 | A2 | (11) International Publication Number: WO 00/14655 (43) International Publication Date: 16 March 2000 (16.03.00) |
| (21) International Application Number: PCT/NO99/00273 (22) International Filing Date: 2 September 1999 (02.09.99) (30) Priority Data: 19984066 3 September 1998 (03.09.98) NO 09/189,626 10 November 1998 (10.11.98) US (71) Applicant: ARENDI AS [NO/NO]; Skovveien 2, N-0257 Oslo (NO). (72) Inventor: HEDLØY, Atle; Terrasseveien 12, N-1320 Stabekk (NO). (74) Agent: TANDBERGS PATENTKONTOR AS; P.O. Box 7085, N-0306 Oslo (NO). | | (81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG). Published <i>Without international search report and to be republished upon receipt of that report.</i> |
| (54) Title: FUNCTION KEY FOR COMPUTER DATA HANDLING (57) Abstract A method, system and computer readable medium for providing a function item, such as a key, button, icon, or menu, tied to a user operation in a computer, whereby a single click on the function item in a window or program on a computer screen, or one single selection in a menu in a program, initiates retrieval of name and addresses and/or other person or company related information, while the user works simultaneously in another program, e.g., a word processor. The click on the function item initiates a program connected to the button to search a database or file available on or through the computer, containing the person, company or address related data, in order to look up data corresponding to what the user types, or partly typed, e.g., name and/or address in the word processor, the correct data from the database, data related to the typed data, e.g., the name of the person, company, or the traditional or electronic address, or other person, or company, or address related data, and alternatively the persons, companies, or addresses, are displayed and possibly entered into the word processor, if such related data exists. | | |

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | | | |
|----|--------------------------|----|--|----|--|----|--------------------------|
| AL | Albania | ES | Spain | LS | Lesotho | SI | Slovenia |
| AM | Armenia | FI | Finland | LT | Lithuania | SK | Slovakia |
| AT | Austria | FR | France | LU | Luxembourg | SN | Senegal |
| AU | Australia | GA | Gabon | LV | Latvia | SZ | Swaziland |
| AZ | Azerbaijan | GB | United Kingdom | MC | Monaco | TD | Chad |
| BA | Bosnia and Herzegovina | GE | Georgia | MD | Republic of Moldova | TG | Togo |
| BB | Barbados | GH | Ghana | MG | Madagascar | TJ | Tajikistan |
| BE | Belgium | GN | Guinea | MK | The former Yugoslav Republic of Macedonia | TM | Turkmenistan |
| BF | Burkina Faso | GR | Greece | | | TR | Turkey |
| BG | Bulgaria | HU | Hungary | ML | Mali | TT | Trinidad and Tobago |
| BJ | Benin | IE | Ireland | MN | Mongolia | UA | Ukraine |
| BR | Brazil | IL | Israel | MR | Mauritania | UG | Uganda |
| BY | Belarus | IS | Iceland | MW | Malawi | US | United States of America |
| CA | Canada | IT | Italy | MX | Mexico | UZ | Uzbekistan |
| CF | Central African Republic | JP | Japan | NE | Niger | VN | Viet Nam |
| CG | Congo | KE | Kenya | NL | Netherlands | YU | Yugoslavia |
| CH | Switzerland | KG | Kyrgyzstan | NO | Norway | ZW | Zimbabwe |
| CI | Côte d'Ivoire | KP | Democratic People's Republic of Korea | NZ | New Zealand | | |
| CM | Cameroon | | | PL | Poland | | |
| CN | China | KR | Republic of Korea | PT | Portugal | | |
| CU | Cuba | KZ | Kazakhstan | RO | Romania | | |
| CZ | Czech Republic | LC | Saint Lucia | RU | Russian Federation | | |
| DE | Germany | LI | Liechtenstein | SD | Sudan | | |
| DK | Denmark | LK | Sri Lanka | SE | Sweden | | |
| EE | Estonia | LR | Liberia | SG | Singapore | | |

Function Key for Computer Data Handling

The present invention is related to a function key,
5 button, icon, or menu, tied to a user operation in a computer,
as defined in the preamble of the claims.

In recent years, with the advent of programs, such as
word processors, spreadsheets, etc., hereinafter called "word
processors" users may require retrieval of information, such as
10 name and address information, etc., for insertion into a
document, such a letter, fax, etc., created with the word pro-
cessor. Typically, the information is retrieved by the user from
an information management source external to the word processor,
such as a database program, contact management program, etc., or
15 from the word processor itself, for insertion into the document.
Examples of such word processors are Word, Notepad, Excel,
Wordpad, WordPerfect, Quattropro, Amipro, etc., and examples of
such information management sources are Access, Outlook, Oracle,
dBase, rBase, Cardfile, etc.

20 However, the information in the database must con-
stantly be updated by the user. This requires the user to learn
how to use and have access to the database. In this case, a
change in the information, such as change in an address or a
name, etc., requires the user of the word processor to implement
25 this change in the database, or alternatively, the change is made
to the database centrally by a database administrator.

The present invention therefor is related to a method,
system and computer readable medium for name and address
handling, hereinafter called "address handling", and more par-
30 ticularly to a touch screen, keyboard button, icon, menu, voice
command device, etc. hereinafter called "button", provided in a
computer program, such as a word processing program, spreadsheet
program, etc., and coupled to an information management source
for providing address handling within a document created by the
35 computer program, thereby providing a method, system and computer
readable medium for address handling within a computer program.

An object of the present invention is to provide a
method, system and computer readable medium for address handling

within a computer program, such as a word processing program, spreadsheet program, etc.

Another object of the present invention is to provide a method, system and computer readable medium for address
5 handling within a computer program, such as a word processing program, spreadsheet program, etc., using an input device, such as a touch screen, keyboard button, icon, menu, voice command device, etc., provided in the computer program and coupled to an information management source, such as a database program,
10 contact management program, etc.

The above and other objects are achieved according to the present invention by providing a novel method, system and computer readable medium for providing a function item, such as a key, button, icon, or menu, tied to a user operation in a
15 computer, whereby a single click on the function item in a window or program on a computer screen, or one single selection in a menu in a program, initiates retrieval of name and addresses and/or other person or company related information, while the user works simultaneously in another program, e.g., a word
20 processor. The click on the function item initiates a program connected to the button to search a database or file available on or through the computer, containing the person, company or address related data, in order to look up data corresponding to what the user types, or partly typed, e.g., name and/or address
25 in the word processor, the correct data from the database, data related to the typed data, e.g., the name of the person, company, or the traditional or electronic address, or other person, or company, or address related data, and alternatively the persons, companies, or addresses, are displayed and possibly entered into
30 the word processor, if such related data exists.

The present invention also includes a computer readable medium storing program instructions by which the method of the invention can be performed when the stored program instructions are appropriately loaded into a computer, and a system for
35 implementing the method of the invention.

A more complete appreciation of the invention and many of the attendant advantages thereof will be readily obtained as the same becomes better understood by reference to the following detailed description when considered in connection with the

accompanying drawings wherein, figure 1 is a flow chart illustrating a method for address handling within a computer program, according to an exemplary embodiment of the present invention, figure 2 is a flow chart illustrating a method for address handling within a computer program, according to another exemplary embodiment of the present invention, figure 3 is a screen shot illustrating the inputting of a name to be searched and an address handling button within a word processor, according to an exemplary embodiment of the present invention, figure 4 is a screen shot illustrating a retrieved address in a word processor, according to an exemplary embodiment of the present invention, figure 5 is a screen shot illustrating the inputting of a name and address to be searched and an address handling button within a word processor, according to an exemplary embodiment of the present invention, figure 6 is a screen shot illustrating an add new contact message window, according to an exemplary embodiment of the present invention, figure 7 is a screen shot illustrating a contact register message window, according to an exemplary embodiment of the present invention, figure 8 is a screen shot illustrating an address missing message window, according to an exemplary embodiment of the present invention, figure 9 is a screen shot illustrating a modify contact's address message window, according to an exemplary embodiment of the present invention, figure 10 is a screen shot illustrating a select a contact address register message window, according to an exemplary embodiment of the present invention, figure 11 is a screen shot illustrating a more detailed mode of registering an additional address for the contact register of figure 9, according to an exemplary embodiment of the present invention, figure 12 is a screen shot illustrating a contact management program window in a full detailed mode, according to an exemplary embodiment of the present invention, figure 13 is a screen shot illustrating an address already in use message window, according to an exemplary embodiment of the present invention, figure 14 is a screen shot illustrating the inputting of a name to be searched and an address handling button within a spreadsheet, according to an exemplary embodiment of the present invention, figure 15 is a screen shot illustrating a retrieved address in a spreadsheet, according to an exemplary

embodiment of the present invention, and figure 16 is a schematic illustration of a general purpose computer for performing the processes of the present invention, according to an exemplary embodiment of the present invention.

5 In an embodiment of the present invention, single button addressing is achieved by providing an input device, such as a touch screen, keyboard, icon, menu, voice command device, etc., hereinafter called "button", in a computer program, such as a word processing program, spreadsheet program, etc.,
10 hereinafter called "word processor", for executing address handling therein.

Accordingly, in a word processor, the button is added and a user types information, such as an addressee's name, or a part of the name, etc. in a document created with the word
15 processor, such as a letter, fax, etc., and then clicks, selects, commands, etc. the button via the appropriate input device, such as a touch screen button, keyboard button, icon, menu choice, voice command device, etc. A program then executes and retrieves the typed information from the document, and searches an
20 information management source, such as a database, file, database program, contact management program, etc., hereinafter called "database" to determine if the information, such as the name or part of the name typed and searched by the program exists in the database. If the program does not find stored information, such
25 as a name, corresponding to the name or part of the name typed, the user is asked by the program whether the information, such as the name that was not found, should be added to the database. In addition, the user may enter any other information besides the name, such as addresses, businesses, telephone numbers, fax
30 numbers, e-mail address, etc., so that this other information can be stored in the database for later use.

If the program finds name(s) and address(es) corresponding to the part of the addressee's name typed, this additional information is automatically entered into the user's
35 word processor, optionally with a confirmation from the user that this is the correct data. If the typed address information does not correspond to data already stored in the database, after clicking on the button, the program, for example, lets the user decide (1) if this is new data (e.g., a new address) for an

existing contact, (2) if the stored data should be changed to what the user just typed, (3) if this is a new contact with the same name as one already entered into the database, or (4) if the typed address is only to be used once, and therefore not to be stored in the database at all. If, later, for example, a name with several address stored in the database is recalled, all addresses for this contact will be displayed, so that the correct address can be selected by the user.

The program may be extended to also store and retrieve other information, such as telephone numbers, fax numbers, e-mail addresses, etc. Once the program recalls the telephone numbers, fax numbers, e-mail addresses, etc., the user can command the program to send e-mails, faxes, etc. Similarly, if the user types in the name of a mailing list, the program create merge letters, group e-mails, etc.

Referring now to the drawings, wherein like reference numerals designate identical or corresponding parts throughout the several views, and more particularly to figures 1 and 2 thereof, there is illustrated flow charts of single button addressing, according to exemplary embodiments of the present invention.

In figure 1, after the user has inserted the address in the word processor, the user commands the button at step 2 and the program analyzes what the user has typed in the document at step 4. At step 6, the program decides what was found in the document and if the program found nothing in the document or what it found was un-interpretable the program goes to step 8 and outputs an appropriate message to the user and then quits at step 16. The program analyzes what the user has typed in the document at step 4, for example, by analyzing (i) paragraph/line separations/formatting, etc., (ii) street, avenue, drive, lane, boulevard, city, state, zip code, country designators and abbreviations, etc., (iii) Mr., Mrs., Sir, Frau, Herr, Madam, Madame, Jr., Sr. designators and abbreviations, etc., (iv) Inc., Ltd., P.C., L.L.C, SA, AS, AB, ASA, AG, designators and abbreviations, etc., and (v) a database of common male/female names, etc.

If the program finds an e-mail address mailing list/category name telephone number or other information, at step

10 an appropriate action is performed by the program and then the program execution quits at step 16. If the program only finds a name or initials, or the like, the program looks up the name in the database at step 12 and at step 18 the program determines
5 what was found. If the program finds more than one possible contact/address match, at step 20 the program displays menu choices to the user to let him choose an appropriate answer. Then at step 22 the program inserts a correct address and name in the document and then at step 16 the program quits execution. If the
10 program finds one match exactly, i.e., one contact with one address, the program inserts the correct address and name in the document at step 22 and then quits execution at step 16. If the program does not find a name in the database, at step 24 the program prompts the user to specify an address and then quits
15 execution at step 16. If the program at step 6 finds a name and an address, at step 14 the name is looked up in the database. Then, at step 26, if no match is found, at step 28 the program inserts an address and a name which are possibly corrected by the user into the database and then quits execution at step 16. If
20 at step 26, the name and address is found, at step 32 the program either takes no action or displays the data for the user to edit. If at step 26, the name is found but not the address, the program prompts the user for a decision at step 30. If the user decides that this another contact with a same name, the program goes to
25 step 28. If the user decides that this is a one time occurrence, no action is taken and the program quits at step 16. If the user decides that the contact has, for example, moved and that this is a new address, at step 34 one of the old addresses for the contact is replaced with the new one and the program quits at
30 step 16. If the user decides that this is an additional address for the contact, at step 36 the additional address is inserted into the database for that contact and execution quits at step 16.

The flowchart shown in figure 2 is similar to the
35 flowchart in figure 1, except for some additional steps which will now be discussed. At step 6, if the program only finds a name or a similar name then the name is looked up in the database at step 12, then at step 18 if the program found more than one possible contact/address match, the program displays choices to

the user to let him choose an address at step 20. Then at step 21 the user decides whether to insert the selected address into the document. If the user does not decide to select the address into the document the program quits execution at step 16. If the user decides to insert the selected address into the document, the program inserts the address and name into the document at step 22 and then quits at step 16.

If the program finds a name and address in the database at step 6, then at step 14 the program looks up the name in the database and at step 26 the program determines what it has found. If the program does not find the name at step 26, at step 27 the program prompts the user for a decision and review and whether to insert the contact and address. If the user does not decide to insert the contact address, the program quits at step 16. If the user decides to insert the contact address, at step 28 the program inserts the address and name which may be possibly corrected by the user or program in the database and then execution quits at step 16.

If at step 26 the program finds a name and not an address, then at step 29 the name is looked up in the database. Then at step 31 the program decides whether this contact has another address. If the contact does not have another address, at step 33 the program prompts the user for a decision and review and whether to add the address. If the user does not want to add the address at step 33, the program quits at step 16. If the user wants to add the address at step 33 because this is an additional address for the contact, at step 36 the address is inserted in the database for the contact and execution quits at step 16.

At step 30, if the user decides that this is another contact with a same name, then the program goes to step 28. If at step 30 the user decides that this is a one time occurrence, then the program quits at step 16. If at step 30, the user decides that the contact has, for example, moved, the program goes to step 34. If at step 30, the user decides that this is an additional address for the contact, at step 36 the program inserts the address in the database for the contact and then quits at step 16.

Various exemplary screen shots which are generated during execution of the program, according to the present

invention, will now be described with reference to figures 3-15 and the examples as follows.

For retrieving an existing address from the database, figure 3 illustrates a starting point in a word processor document, such as a Word document, wherein the user has typed a name 40. The user hits the button 42, for example, marked "OneButton" and the program according to the present invention retrieves the name 40 from the document, searches a database for the name 40, and inserts the retrieved address 44 associated with the name 40 into the document as shown in, for example, figure 4.

The above example corresponds to steps 2, 4, 6, 12, 18, 22 and 16 in the flow charts of figures 1 and 2.

For adding a new contact to the database figure 5 illustrates a starting point in word processor document, such as a Word document, wherein the user has typed a name and address of a new contact 46. The user commands the button 42, for example, marked "OneButton," and the program according to the invention retrieves the new contact 46 from the document, searches a database for the name of the new contact 46 and generates a screen as shown in, for example, figure 6. This screen includes a message 50 informing the user that the new contact does not exist in the database, a message 52 including the address retrieved from the document, an address type selection 54, such as home, business, etc., and "OK", "Details", and "Cancel" buttons 56, 58, and 60, respectively.

At this point, the user can cancel the operation by commanding the "Cancel" button 60, ask the program to store data in the database and return to the document by commanding the "OK" button 56, or check details before storing data into the database by commanding the "Details" button 58. If the user commands the Details button 58, as shown in, for example, figure 7, a message screen is provided so that the user can review and edit data 62 and the selection 54, store the data 62 and 54 in the database by commanding a "Add and Choose" button 64, see more options by commanding an "Options" button 66, or cancel the operation by commanding the "Cancel" button 60.

The above example corresponds to steps 2, 4, 6, 14, 26, 28 and 16 in the flow chart of figure 1 and steps 2, 4, 6, 14, 26, 27, 28 and 16 in the flow chart of figure 2.

For trying to retrieve an existing address, but where
5 contact is not in the database, figure 3 illustrates a starting point in word processor document, such as a Word document, wherein the user has typed a name of a contact 40. The user commands the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the name 40
10 from the document, searches a database for the name of the contact 40 and generates a screen as shown in, for example, figure 8. This screen includes a message 68 informing the user that the contact does not exist in the database and to specify an address, and "OK" buttons 56. At this point when the user
15 commands the "OK" button 56, the user returns to the document so that the contact's address can be included as in example 2 above.

The above example corresponds to steps 2, 4, 6, 12, 18, 24 and 16 in the flow charts of figures 1 and 2.

20 For adding a new address for an existing contact in a short version, figure 4 illustrates a starting point in word processor document, such as a Word document, wherein the user has typed a name and new address of an existing contact 44. The user commands the button 42, for example, marked "OneButton," and the
25 program according to the present invention retrieves the existing contact 44 from the document, searches a database for the name of the existing contact 44 and generates a screen as shown in, for example, figure 9. This screen includes a message 70 informing the user that the contact already exists in the database
30 with an existing address, a message 72 including the existing address, add new contact with same name selection 74, change existing address selection 76, use existing address in document selection 78, add the new address to contact selection 80, the address type selection 54, such as home, business, etc., and the
35 "OK", "Details", and "Cancel" buttons 56, 58, and 60 respectively. At this point, the user may select one of the four options 74-80, and command the "OK" button 56 to execute the selected options. The user can also cancel the operation by commanding the "Cancel" button 60, or check details before

storing data into the database by commanding the "Details" button 58.

The above example corresponds to steps 2, 4, 6, 14, 26, 28, 30, 34, 36, and 16 in the flow chart of figure 1 and steps 2, 4, 6, 14, 26, 29, 31, 30, 28, 34, 36, and 16 in the flow chart of figure 2.

For selecting between several possible matching addresses, figure 3 illustrates a starting point in a word processor document, such as a Word document, wherein the user has typed a name and possibly address of at least one existing contact 40. The user commands the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the existing contact 40 from the document, searches a database for the name of the existing contact 40 and generates a screen as shown in, for example, figure 10. This screen includes a message informing the user that the name corresponds to several addresses and possible contacts which already exist in the database, with existing contacts and addresses for selection 82, a message 84 including the full name and address for the contact that the user selects in 82, the "Options" button 66, a "Choose" button 86, a "Full details" button 88, a "More>>>" button 90, and the "Cancel" button 60. The above screen indicates to the user that at least one contact with the same name exists, and that there are more than one addresses and/or contacts that match.

At this point, the user may command the "Choose" button 86 to use the selected address and return to the document, or the user may command the "More>>>" button 90 to view how the program interpreted what the user typed in the word processor, and possibly change this data, wherein the program generates an updated screen as shown in, for example, figure 11. The updated screen includes the data 62 which displays the name typed in the word processor as interpreted by the program, address fields, and the fields for the address type selection 54, such as home, business, etc., which may be changed by the user before the program stores it in the database, the "Add and Choose" button 64, a "<<<Less" button 90 corresponding to the "More>>>" button 90 for returning to the screen of figure 10, and an "Add this address to the selected contact above" button 92. The user might

then command the "Add this address to the selected contact above" button 92 and the result in the word processor is illustrated in figure 4. The user can also cancel the operation by commanding the "Cancel" button 60, or command the "Add and Choose" button 64 to add this name and address as a new contact and address, or open the database before storing data into the database by commanding a "Full details" button 88 as will be later described.

The above example corresponds to steps 2, 4, 6, 12, 18, 20, 22, and 16 in the flow chart of figure 1 and steps 2, 4, 6, 10 12, 18, 20, 21, 22, and 16 in the flow chart of figure 2.

For adding a new address for an existing contact in a long version figure 4 illustrates a starting point in word processor document, such as a Word document, wherein the user has typed a name and new address of an existing contact 44. The user 15 commands the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the existing contact 44 from the document, searches a database for the name of the existing contact 44 and generates a screen as shown in, for example, figure 9. As previously described, the screen 20 includes a message 70 informing the user that the contact already exists in the database with an existing address, and the user may command the "Details" button 58 to see the details of the new address for potentially modify the details before they are stored in the database and the program generates a screen as shown in, 25 for example, figure 10. From this screen, the user may choose to use another address than the one he typed, and return to the document, or the user may command the "Full details" button 88 to enter a database program, such as Outlook, directly as shown in, for example, figure 12. In figure 12, the database program, 30 such as Outlook, may include portions 94-104 for allowing the user to modify various pieces of data before they are stored in the database.

Alternatively, in the screen shown in figure 10, the user may command the "More>>>" button 90 at which time the 35 program generates the screen as shown in, for example, figure 11 and as previously described. In this screen, the user might then command the Add this address to the selected contact above button 92. If the address typed is already in use, the program generates a screen including a message 106, and "Yes" and "No" buttons, 108

and 110, respectively, as shown in, for example, figure 13. If the user hits the "Yes" button 108 the program overwrites the contact address with the address specified by the user (e.g., if the contact has moved) and the result in the word processor is shown in, for example, figure 4.

The above example corresponds to steps 2, 4, 6, 12, 14, 26, 28, 30, 34, 36, and 16 in the flow chart of figure 1 and steps 2, 4, 6, 12, 14, 26, 29, 31, 30, 28, 34, 36, and 16 in the flow chart of figure 2.

For a spreadsheet application figure 14 illustrates a starting point in word processor document, such as an Excel spreadsheet, wherein the user has typed a name 112. The user hits the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the name 112 from the spreadsheet, searches a database for the name 112, and inserts the retrieved address 114 into the spreadsheet as shown in, for example, figure 15. Accordingly, the above examples apply not only to word processor documents, such as Word documents, etc., but to other word processor documents, and spread sheets, such as Excel spreadsheets, etc.

The above example corresponds to steps 2, 4, 6, 12, 18, 22 and 16 in the flow charts of figures 1 and 2.

Figure 16 is a schematic illustration of a computer system for implementing the single button addressing according to the present invention. A computer 200 implements the method of the present invention, wherein the computer includes, for example, a display device 202, such as a conventional display device or a touch screen monitor with a touch-screen interface, etc., a keyboard 204, a pointing device 206, a mouse pad or digitizing pad 208, a hard disk 210, or other fixed, high density media drives, connected using an appropriate device bus (e.g., a SCSI bus, an Enhanced IDE bus, an Ultra DMA bus, a PCI bus, etc.), a floppy drive 212, a tape or CD ROM drive 214 with tape or CD media 216, or other removable media devices, such as magneto-optical media, etc., and a mother board 218. The mother board 218 includes, for example, a processor 220, a RAM 222, and a ROM 224 (e.g., DRAM, ROM, EPROM, EEPROM, SRAM, SDRAM, and flash RAM, etc.), I/O ports 226 which may be used to couple to external devices, networks, etc., (not shown), and optional special

purpose logic devices (e.g., ASICs) or configurable logic devices (e.g., GAL and re-programmable FPGA) 228 for performing specialized hardware/software functions, such as sound processing, image processing, signal processing, neural network
5 processing, object character recognition (OCR) processing, etc., a microphone 230, and a speaker or speakers 232.

As stated above, the system includes at least one computer readable medium, or alternatively, the computer readable medium may be accessed through various paths, such as networks,
10 internet, drives, etc. Examples of computer readable media are compact discs, hard disks, floppy disks, tape, magneto-optical disks, PROMs (EPROM, EEPROM, flash EPROM), DRAM, SRAM, SDRAM, etc. Stored on any one or on a combination of computer readable media, the present invention includes software for controlling
15 both the hardware of the computer 200 and for enabling the computer 200 to interact with a human user. Such software may include, but is not limited to, device drivers, operating systems and user applications, such as development tools. Such computer readable media further includes the computer program product of
20 the present invention for performing any of the processes according to the present invention, described above (see, e.g., figures 1-15). The computer code devices of the present invention can be any interpreted or executable code mechanism, including but not limited to scripts, interpreters, dynamic link libraries,
25 Java classes, and complete executable programs, etc.

The invention may also be implemented by the preparation of application specific integrated circuits or by interconnecting an appropriate network of conventional component
30 circuits, as will be readily apparent to those skilled in the art.

Address handling, according to this invention, is a significant simplification relative to existing methods, and requires little or no training on the part of a user, as correct addresses are retrieved with a minimal number of user commands,
35 "clicks", keystrokes, etc. In addition, a program according to the present invention, can be programmed and created in most existing programming languages and be connected to most modern word processors. Therefore, according to the present invention, the process of creating and updating records in an address

database is significantly simplified, since this may now be performed directly from the word processor.

Although the present invention is defined in terms of word processing documents, such as Word documents and Excel
5 spreadsheets, the present invention is applicable to all types of word processing documents, such as Notepad, Wordpad, WordPerfect, Quattropro, Amipro, etc., as will be readily apparent to those skilled in the art.

Although the present invention is defined in terms of
10 information management or database programs, such as Outlook, etc., the present invention is applicable to all types of information management or database programs, such as Access, Oracle, dBase, rBase, Cardfile, including "flat files", etc., as will be readily apparent to those skilled in the art.

15 Although the present invention is defined in terms of providing an input device, such as a button 42 in a word processor for address handling therein, the present invention may be practised with all types of input devices, such as a touch screen, keyboard button, icon, menu, voice command device, etc.,
20 as will be readily apparent to those skilled in the art.

Although the present invention is defined in terms of a program retrieving information from a document before searching a database, the user may select the information in the document to be searched by the program in the database (e.g., by
25 highlighting, selecting, italicizing, underlining, etc.), as will be readily apparent to those skilled in the art.

Although the present invention is defined in terms of a program retrieving a name or portion thereof from a document before searching a database, the program may retrieve an address
30 or portion thereof from the document before searching the database and insert, correct, complete, etc., the retrieved address based on the information found in the database corresponding to the retrieved address or portion thereof, as will be readily apparent to those skilled in the art.

35 Furthermore the number of names or addresses found with each new character typed may be displayed on the screen during typing, starting with the first character typed, the user thereby at any time having the possibility to display on the screen the names or addresses found for further analysis before selecting.

In this way the user may decide how many characters be typed before transfer to the screen and also to a certain degree capture names or addresses typed misspelled by the user or stored misspelled.

5 The user may decide by a corresponding command button whether the search should be limited to the first word only or to any single word in a name or an address, such as selecting "High Noon Adventure" by typing only "High", "High Noon" or "High Noon Adventure", or optionally also by typing "Noon" or
10 "Adventure".

15

20

25

30

35

P a t e n t C l a i m s

5 1. Function key, button, icon, or menu, tied to a user operation in a computer, such one single click on a button in a window or program on a computer screen, or one single selection in a menu in a program, **CHARACTERIZED IN** the function initiates retrieval of name and addresses and/or to other person or company
10 related information, while the user works in another program, e.g., a word processor, that a click on the button leads to the program connected to the button searches in a database or file available on or through the computer containing the person, company or address related data, in order to look up data
15 corresponding to what the user types, being it the full or part of the name or address, e.g., name and/or address in the word processor, that correct data from the database, data related to the typed data, e.g., the name of the person, company, or the address, alternatively the persons, companies, or addresses, is
20 displayed and possibly entered into the word processor by the user, if such related data exists and being correct.

 2. Button, according to Claim 1, **CHARACTERIZED IN** that if the data typed by the user does not correspond to any previously stored data in the database, the user will be given
25 the opportunity to select if new data should be entered in the database, or existing data in the database should be changed to reflect the new data, or existing data in the database shall be used instead of what the user had typed, or not to take any action.

30 3. Button according to Claim 1, **CHARACTERIZED IN** that if the typed data is incomplete or erroneous, the program connected to the button will, alone, or with user assistance, complete and/or correct the address, based on algorithms and data available in and for the function for this purpose.

35 4. Button according to Claim 3, **CHARACTERIZED IN** the incomplete or erroneous information is that the zip code, town, state, or street name or abbreviations thereof, and that the functionality finds and or corrects this incomplete or erroneous

information, based on algorithms and data available in and for the function for this purpose.

5. Button according to Claim 1, **CHARACTERIZED IN** that if the previously typed data refers to a category or group or persons or companies, and that the functionality will assist the user to create and/or send documents to all members of the category or group.

6. Button according to Claim 1, **CHARACTERIZED IN** that if the database has access possibilities it make changes, alternatively other data handling, directly in the database.

7. Button according to Claim 1, **CHARACTERIZED IN** that if the information to be fetched can be selected in advance, e.g., postal address, telefax number, e-mail address, etc.

8. Button according to Claim 1, **CHARACTERIZED IN** the number of the names or addresses found with each new character typed being displayed on the screen during typing, starting with the first character typed, the user thereby at any time having the possibility to command display of the found names or addresses on the screen.

9. Button according to Claim 1, **CHARACTERIZED IN** activation or not of a command button decides whether the search is limited to the first word of the names or addresses or to any single word of the names or addresses.

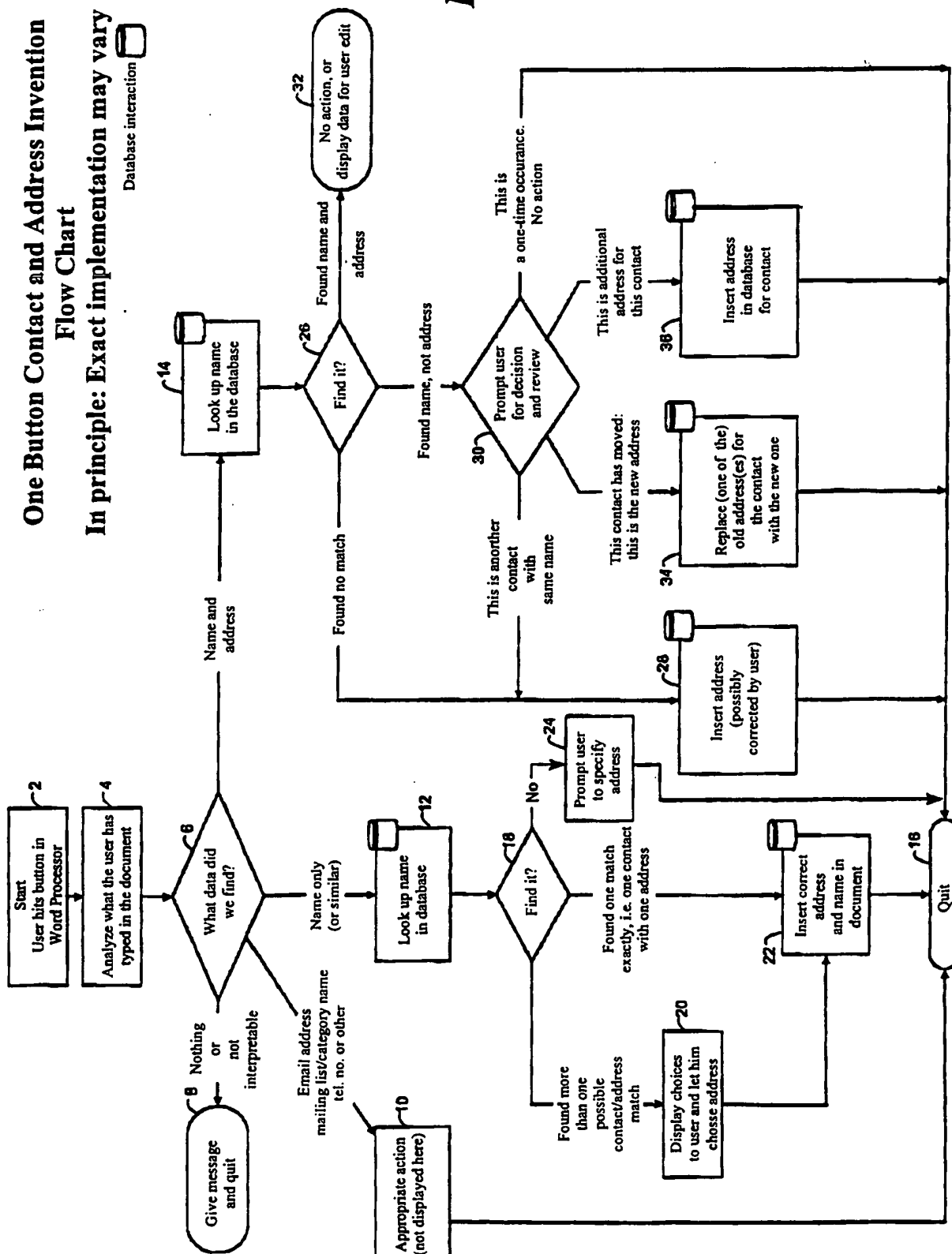
25

30

35

1/13

Fig. 1

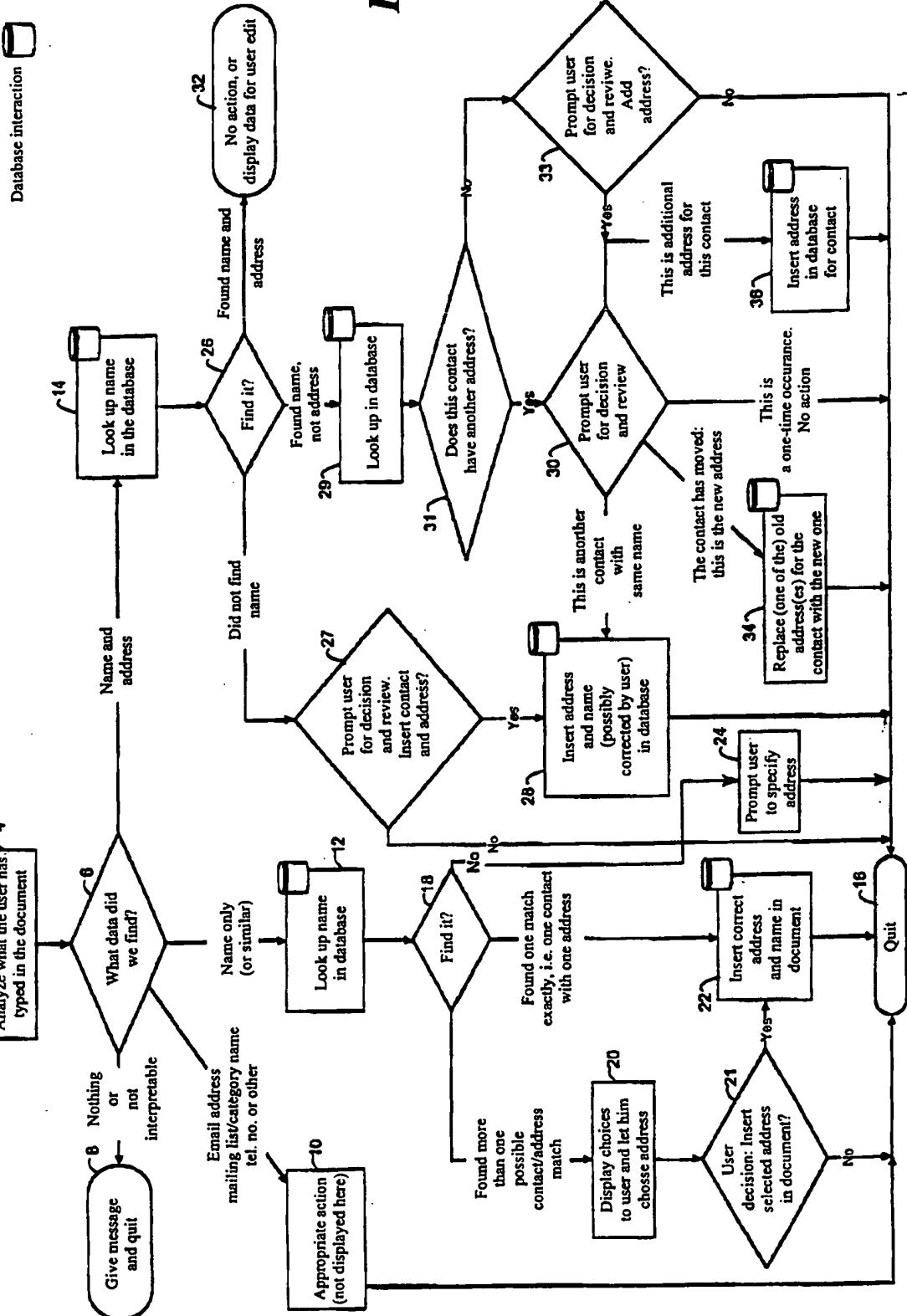


2/13

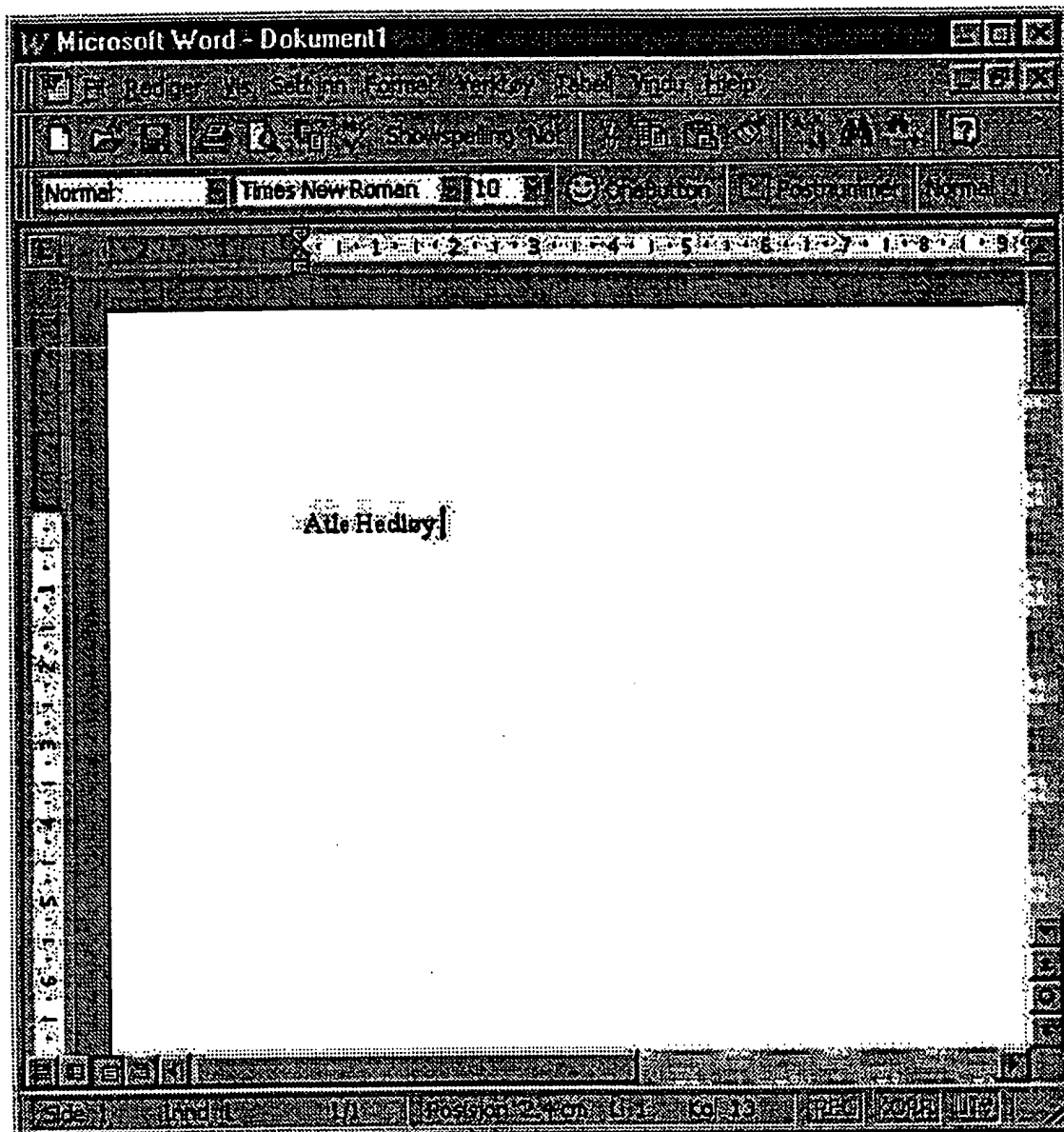
Fig. 2

One Button Contact and Address Invention Flow Chart

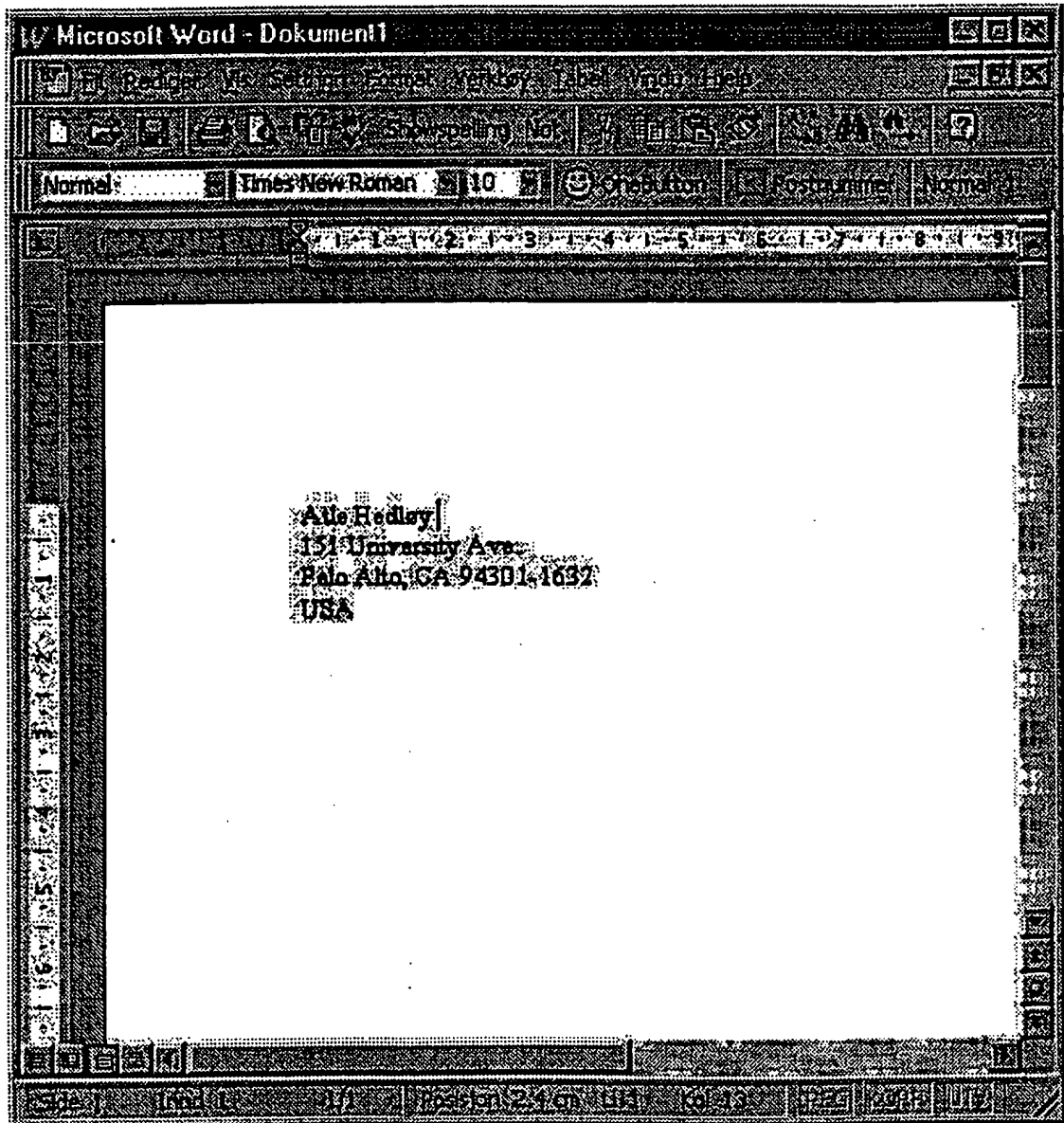
In principle: Exact implementation may vary



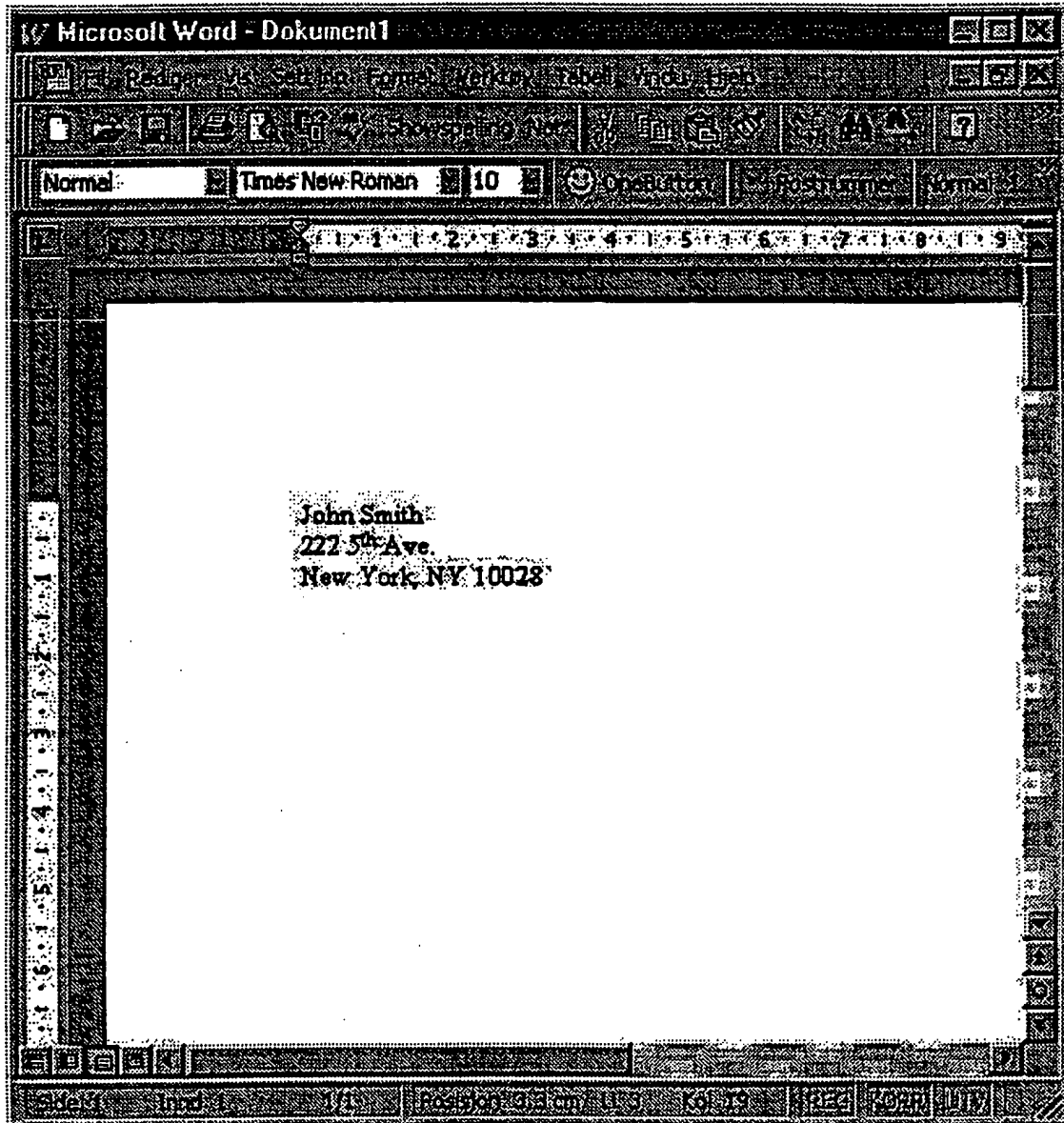
3/13

*Fig. 3*

4/13

*Fig. 4*

5/13

*Fig. 5*

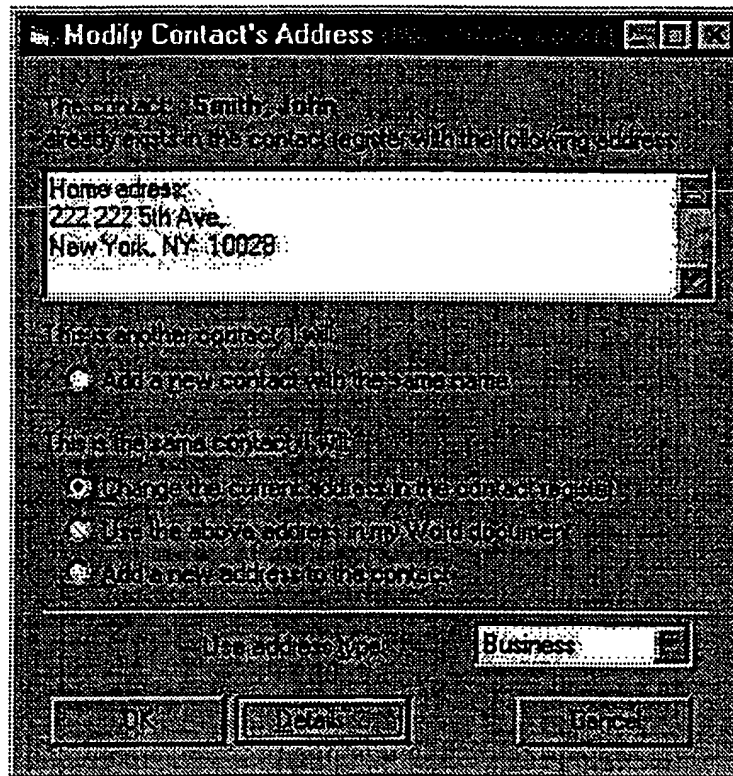
6/13

Arendt OneButton Contact Register

| | | |
|---|-------------|-----------------|
| Name | Home | Address |
| First | 222 5th Ave | Street |
| Middle | New York | City |
| Last | NY | State/Province |
| Suffix | 10028 | ZIP/Postal Code |
| Company | | Country |
| <input type="button" value="Add and Choose"/> | | |
| <input type="button" value="Options"/> | | |
| Date or an test | | |

Fig. 7

7/13

*Fig. 9*

8/13

Arendi OneButton Contact Register

Enter addresses into the same panel

| Name | Address | Phone | Zip | City | State | Country |
|-------------|---|-------|-----|------|-------|---------|
| Alle Hedley | 113 Terrace Street New York, NY 12181-4292 United States of America | | | | | |
| Home | 113 Terrace Street New York, NY 12181-4292 United States of America | | | | | |

Name: Alle Hedley Address: 113 Terrace Street, New York, NY 12181-4292, United States of America

Home: 113 Terrace Street, New York, NY 12181-4292, United States of America

Phone: Add and Remove

Zip: Add and Remove

City: Add and Remove

State: Add and Remove

Country: Add and Remove

Options: Add and Remove

Cancel

Fig. 10

9/13

Fig. 11

Arendi OneButton Contact Register

Editing addresses for the same device

| Name | Address type | Address | Country |
|-----------------|--------------|---------------------------|--------------------------|
| 111 Hedy's line | Business | 111 Terrace street | United States of America |
| Home | Home | 113 113 Jacobb Aal street | 12191-1292 New York |

Name: Address: Country:

113 Terrace street
New York NY 12191-1292
United States of America

Name: Address type: Address: Country:

Home: 151 University Ave.
Photo:
EAI:
99011-1532
USA:

Add and Edit address to the address list on the above

Options: Cancel:

Deleted:

10/13

Fig. 12

Alle Hedley - Contact

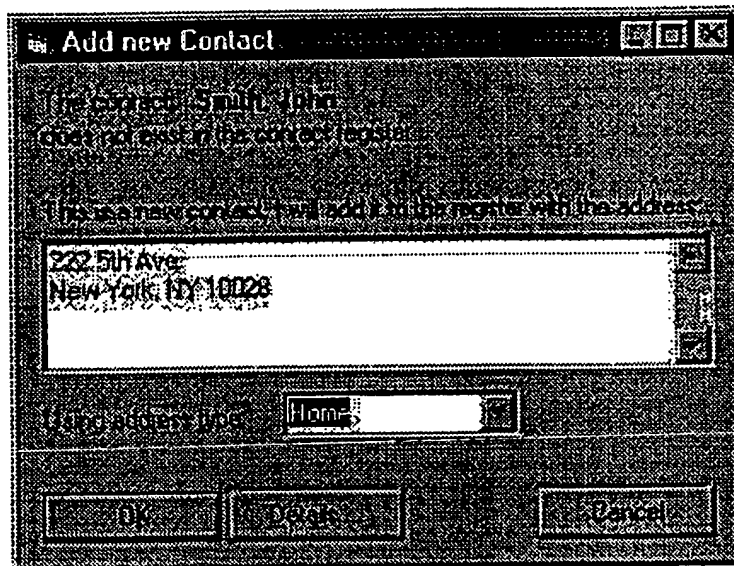
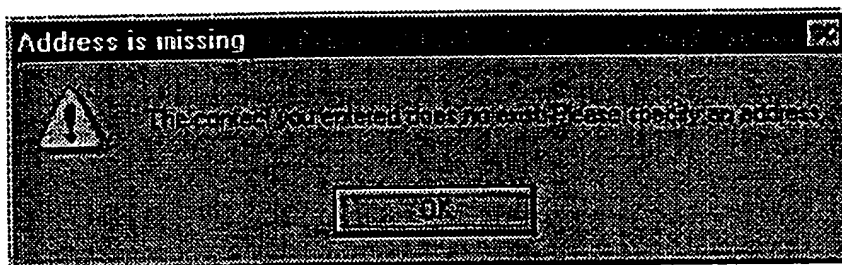
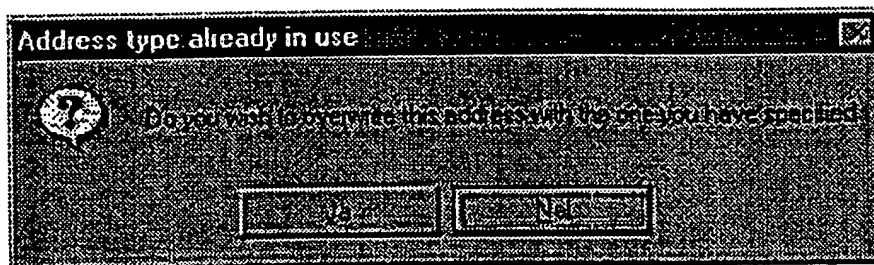
☐ New ☐ Edit ☐ Delete ☐ Print ☐ Find ☐ Add ☐ Remove ☐ Refresh ☐ Help

☐ Name ☐ Address ☐ Phone ☐ Email ☐ Notes

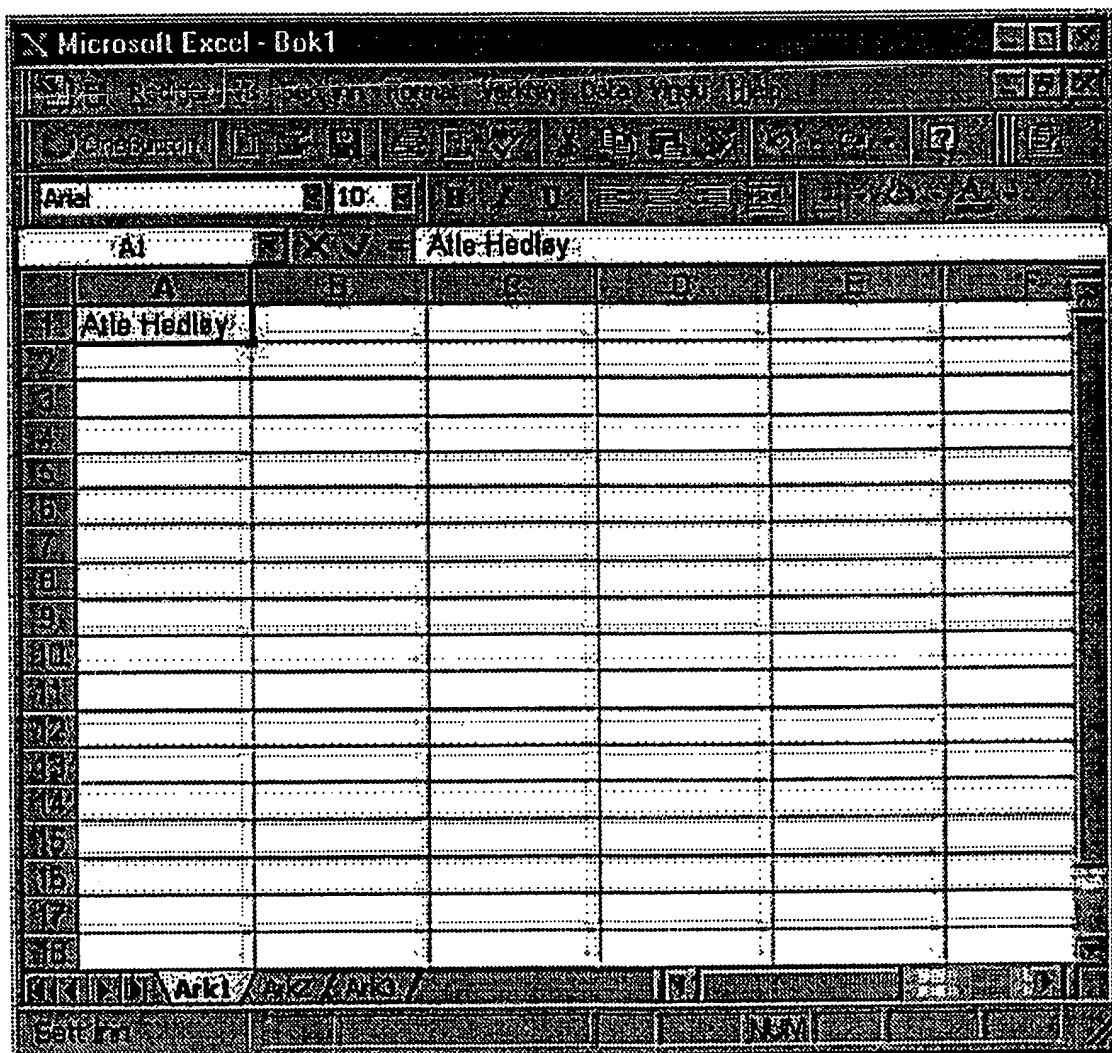
☐ Name: Alle Hedley
☐ Address: 113 Terrace Street
 New York, NY 12191-4292
 United States of America
☐ Phone:
☐ Business: ☐ Home: ☐ Business Fax: ☐ Mobile:
☐ E-mail:

☐ Save ☐ Cancel ☐ OK ☐ Help

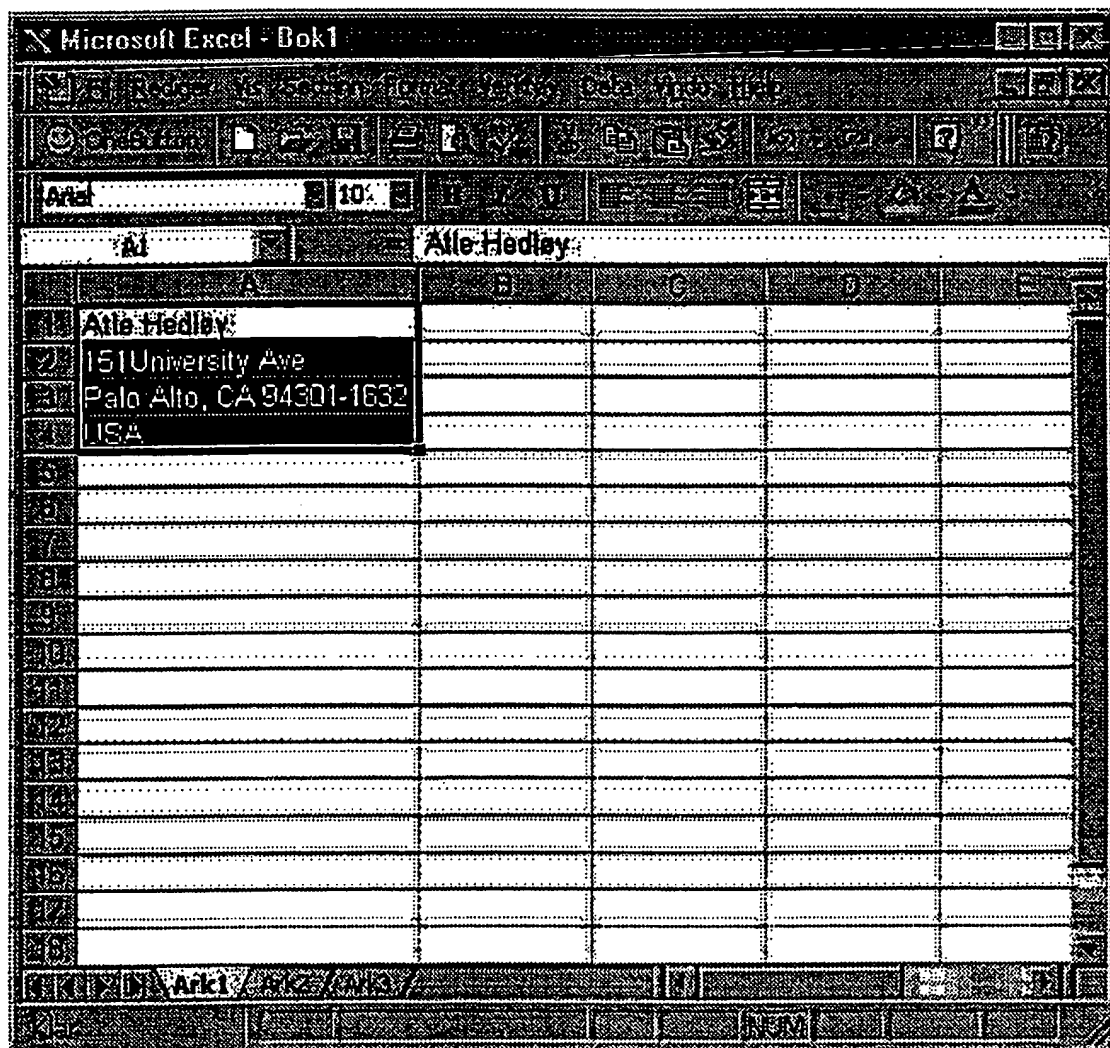
11/13

*Fig. 6**Fig. 8**Fig. 13*

12/13

*Fig. 14*

13/13

*Fig. 15*